ESPOCH Congresses: The Ecuadorian Journal of S.T.E.A.M.
Volume 2, Issue no. 5, DOI 10.18502/espoch.v2i5.11734
Production and Hosting by Knowledge E

Research Article

Vehicle and Pedestrian Detection in Traffic Videos Using Convolutional Neural Networks

Detección de Vehículos y Peatones en Videos de Tráfico, Usando Redes Neuronales Convolucionales

P, Arroba-Villacis[1]*, W, Maiza Pérez[1], C, Carrión-Paladines[1], F, Revelo-Aguilar[2]

[1]Instituto Superior Pedagógico Martha Bucaram de Roldós Bilingüe – Nueva Loja, Ecuador
[1]Universidad Técnica de Ambato, Ambato- Ecuador

ORCID
P, Arroba-Villacis: https://orcid.org/0000-0001-5601-0863

I INTERNATIONAL SCIENTIFIC CONGRESS OF INNOVATION, SCIENCE AND TECHNOLOGY ALIVE AMAZON (I CTAV 2021)

Corresponding Author: P, Arroba-Villacis; email: patricio.javi@hotmail.com

Published: 1 September 2022

Production and Hosting by Knowledge E

© P, Arroba-Villacis et al. This article is distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use and redistribution provided that the original author and source are credited.

Abstract

One of the major applications of computer vision is the analysis of the traffic scene on the road, and how pedestrian traffic affects traffic in general. Road sizes and traffic signals must constantly adapt. Counting and classifying vehicles and pedestrians at an intersection is an exhausting task, and despite the use of traffic control systems, human interaction is very necessary to perform such a task. The object of study of Deep Learning is to try to solve problems that require artificial intelligence. Artificial intelligence has been working in this field for years, with different approaches and algorithms. It has achieved an important emergence in the recognition of patterns in images and videos using these techniques, to the point of surpassing human capacity in some problems. An important factor in this development is the ability to process large volumes of information in applications, which has resulted in the devices used for this purpose, such as GPU's and multi-core CPU's, requiring a large amount of power to operate. For the development of the application of vehicle and pedestrian detection in traffic videos, YOLO V3 was used, which is a neural network model of the latest generation of real-time objects.

Keywords: *yoloV3, Deep Learning, Convolucional Network.*

Resumen

Una de las mayores aplicaciones de la visión por computadora es el análisis de la escena de tráfico en la carretera, y cómo el tráfico de peatones afecta al tráfico en general. Los tamaños de las carreteras y las señales de tráfico deben adaptarse constantemente. Contar y clasificar vehículos y peatones en una intersección es una tarea agotadora y, a pesar del uso de sistemas de control de tráfico, la interacción humana es muy necesaria para realizar dicha tarea. El objeto de estudio de Deep Learning, es intentar resolver problemas que requieren inteligencia artificial. La inteligencia artificial ha trabajado en este campo durante años, con diferentes enfoques y algoritmos. Ha logrado un surgimiento importante en el reconocimiento de patrones en imágenes y videos usando estas técnicas, hasta el punto de superar la capacidad humana en algunos problemas. Un importante factor de este desarrollo es la capacidad de procesar grandes volúmenes de información en aplicaciones, lo que ha dado como resultado que los dispositivos utilizados para este propósito, como GPU's y CPU's multinúcleo, requieran una gran cantidad de energía para operar. Para el desarrollo de la aplicación de Detección de vehículos y peatones en videos de tráfico, fue utilizado YOLO V3, que es un modelo de red neuronal de la última generación de objetos en tiempo real.

Palabras Clave: *yoloV3, Aprendizaje profundo, Red convolucional.*

OPEN ACCESS

1. Introducción

Como consecuencia del desarrollo de los sistemas de visión por computadora, ha surgido la necesidad de contar con sistemas de monitoreo y vigilancia para situaciones específicas, que superen las limitaciones del sistema de percepción humano en el sentido de atención, vigilancia y monitoreo [1,10,15,17]. La detección de peatones por medio de procesamiento de imágenes por computadora es un tema que estimula un gran interés en la comunidad científica. Los principales trabajos pueden dividirse en función de las técnicas utilizadas para extraer información [2,11,13,16]. Una de las mayores aplicaciones de estos sistemas son medir el fujo vehicular y peatonal, para maximizar el uso de los semáforos de tal forma que el tiempo de espera dependa de las necesidades del tráfico [1,12,14,18]. El Deep learning permite modelos computacionales que están compuestos de varias capas de procesamiento para aprender representaciones de datos con múltiples niveles de abstracción. Estos métodos han mejorado el estado del arte en reconocimiento de voz, reconocimiento de objetos visuales, detección de objetos y muchos otros dominios como el descubrimiento de fármacos y el genoma humano [6,7,8]. El Deep learning descubre una estructura compleja en grandes conjuntos de datos utilizando el algoritmo de retro propagación. Las redes convolucionales han producido avances en el procesamiento de imágenes, video y audio, mientras que las redes recurrentes han producido avances en el tratamiento de datos secuenciales como texto y voz [3,9,19]. La forma más común de aprendizaje profundo es supervisada. Un ejemplo de esto sería construir un sistema de clasificación de imágenes que contenga una casa, un coche, una persona y una mascota, primero se recoge un gran número de imágenes cada una etiquetada con su categoría. Con estos datos se produce un entrenamiento y se genera un vector de puntuaciones una para cada categoría, luego de esto se computa una función objetiva que mida el error de salida de puntuaciones y genere un patrón de puntuaciones para cada categoría [3,20,21]. Se ajusta el vector de peso, el algoritmo computa un vector de gradiente que, para cada peso, indica por qué cantidad el error aumentaría o disminuiría si el peso estuvo aumentado por una cantidad minúscula. El vector de peso es entonces ajustado en la dirección opuesta al vector de gradiente. Este procedimiento sencillo normalmente encuentra un conjunto de pesos sorprendentemente rápidos cuando se compara con técnicas de optimización. Después de entrenar, el rendimiento del sistema está medido en un conjunto diferente de ejemplo un conjunto de prueba. Esto sirve para probar la capacidad de generalización del algoritmo y su capacidad de producir respuestas sensatas en entradas nuevas que nunca ha visto durante el entrenamiento [3,22,23].

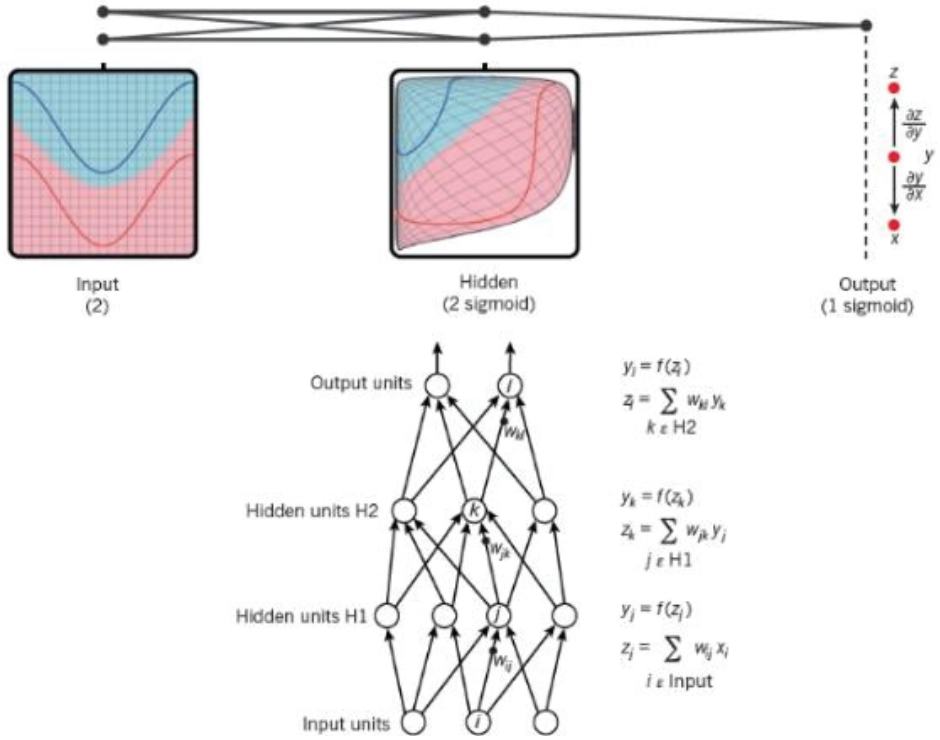

Figure 1

Multilayer neural networks and backpropagation. Fuente: Rusk N. Deep learning. Nature Methods. 2016.

En la figura 1 se muestra una red neuronal multicapa que puede distorsionar el espacio de entrada para hacer que las clases de datos sean linealmente separables. Se observa como una cuadrícula regular en el espacio de entrada también se transforma mediante unidades ocultas. Este es un ejemplo ilustrativo con solo dos unidades de entrada, dos unidades ocultas y una unidad de salida, pero las redes utilizadas para el reconocimiento de objetos o el procesamiento del lenguaje natural contienen decenas o cientos de miles de unidades. [3,4]. Las ecuaciones utilizadas para calcular el paso directo en una red neuronal con dos capas ocultas y una capa de salida se puede observar en la figura 1, cada una de las cuales constituye un módulo a través del cual se puede propagar hacia atrás los gradientes. En cada capa oculta calculamos la derivada de error con respecto a la salida de cada unidad, que es una suma ponderada de las derivadas de error con respecto a las entradas totales a las unidades en la capa anterior [3]. Las redes neuronales convolucionales están diseñadas para procesar datos que vienen en forma de múltiples matrices, por ejemplo, una imagen en color compuesta por tres matrices 2D que contienen intensidades de pixeles en los tres canales de color como se muestra en la figura 2. Las unidades en una capa convolucional están organizadas en mapas de características, dentro de los cuales cada unidad está conectada a parches locales en los mapas de características de la

capa anterior a través de un conjunto de pesos llamado banco de filtros. Todas las unidades en un mapa de características comparten el mismo banco de filtros. Primero, en los datos de matriz, como las imágenes, los grupos locales de valores a menudo están altamente correlacionados, formando motivos locales distintivos que se detectan fácilmente [3,24,25].

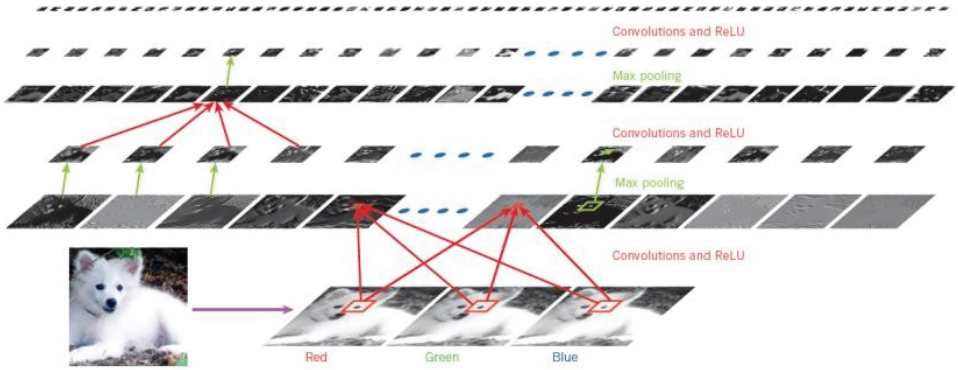

Figure 2

Inside a convolutional networkcite. Fuente: Rusk N. Deep learning. Nature Methods. 2016.

A continuación, se presenta trabajos relacionados:

People Detection System Using YOLOv3 Algorithm. In 2020 10th IEEE international conference on control system, computing and engineering.- Este trabajo presenta una red neuronal convolucional (CNN) que se entrena utilizando un modelo de (YOLOv3) en Google Colaboratory para procesar las imágenes dentro de una base de datos y ubicar con precisión a las personas dentro de las imágenes. YOLOv3 divide la imagen en regiones y predice cuadros delimitadores y predice las probabilidades para cada región. Estos cuadros delimitadores son ponderados por las probabilidades proyectadas y, finalmente, el modelo puede realizar su detección en función de los pesos finales. Una vez entrenada, la red neuronal puede generar con éxito los datos de prueba y lograr una precisión promedio promedio (mAP) de 78,3 % y una pérdida promedio final de 0,6 además de detectar con confianza a las personas dentro de las imágenes [21].

Pedestrian detection based on TensorFlow YOLOv3 embedded in a portable system adaptable to vehicles. In International Conference on Development and Application Systems.- El propósito de esta investigación es demostrar y proponer soluciones viables que ayudarán a los conductores a practicar un estilo de conducción eficiente, seguro y sin incidentes. En este trabajo se presenta un prototipo en desarrollo que puede evitar diversos eventos de tránsito, analizando el sistema y alertando al conductor sobre intenciones peatonales, marcando cada detección por separado de acuerdo

al grado de peligro que constituye tanto para el conductor, como para los peatones [18].

Real-time detection of vehicle and traffic light for intelligent and connected vehicles based on YOLOv3 network. In 5th International Conference on Transportation Information and Safety.- En este documento, se establece un nuevo conjunto de datos de vehículos y semáforos y se presenta un modelo de detección en tiempo real de vehículos y semáforos basado en la red You Look Only Once (YOLO). YOLOv3 propone un método de entrenamiento conjunto para la clasificación y detección de objetivos, con el objetivo de equilibrar la precisión y la velocidad de detección. A través del análisis experimental de las imágenes medidas en el entorno urbano, se demuestra que el modelo diseñado no solo puede satisfacer los requisitos en tiempo real, sino también mejorar la precisión de la detección de vehículos y semáforos [17].

Data-driven based tiny-YOLOv3 method for front vehicle detection inducing SPP-net.- Con el fin de resolver el problema de la baja tasa de reconocimiento y el bajo rendimiento en tiempo real de la detección de vehículos en un entorno vial complejo, se propone un algoritmo de detección basado en tiny-YOLOv3 mejorado, Los resultados experimentales muestran que la precisión promedio del algoritmo mejorado en los conjuntos de datos de KITTI es del 91,03 %, que es un 7,12 % más alta que la de tiny-YOLOv3. Y la velocidad de detección de la red mejorada es de 144 cuadros/s, lo que cumple con los requisitos en tiempo real [16].

2. Materiales y Métodos

Las pruebas se realizaron con una PC de las siguientes características.

1. Procesador Intel Core i7 – 7700hq
2. Tarjeta gráfica NVIDIA GeForce GTX 1060
3. 16 GB Memoria DDR4.
4. Disco duro de 256 GB SSD.

YOLOV3.- YOLO es un sistema de detección de objetos en tiempo real de última generación, YOLOv3 es extremadamente rápido y preciso. En mAP medido a .5 IOU YOLOv3 está a la par con la pérdida focal pero aproximadamente 4 veces más rápido como se observa en la figura 3. Además, puede intercambiar fácilmente entre velocidad y precisión simplemente cambiando el tamaño del modelo, no se requiere reentrenamiento [4,5].

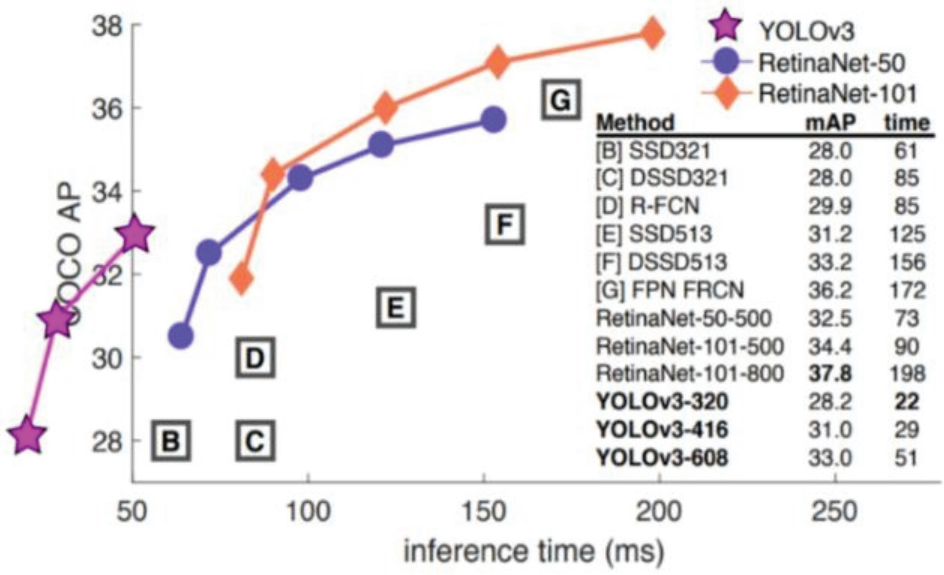

Figure 3

Comparative table of YOLOv3 velocities. Fuente: Simple online and realtime tracking. In IEEE international conference on image processing, 2016.

Cómo funciona YOLOv3.- YOLOv3 aplica una única red neuronal a la imagen completa. Esta red divide la imagen en regiones y predice cuadros delimitadores y probabilidades para cada región [4]. Estos cuadros delimitadores están ponderados por las probabilidades predichas. El modelo tiene varias ventajas sobre los sistemas basados en clasificadores. Analiza la imagen completa en el momento de la prueba para que sus predicciones informadas por el contexto global de la imagen. También hace predicciones con una única evaluación de red a diferencia de sistemas como R-CNN que requieren miles para una sola imagen. YOLOv3 se ejecuta significativamente más rápido que otros métodos de detección con un rendimiento comparable [4,24]. La red predice 4 coordenadas para cada cuadro delimitador, tx, ty, tw, th. Si la celda está desplazada desde la esquina superior izquierda de la imagen por (cx; cy) y el cuadro delimitador anterior tiene ancho y altura pw, ph, entonces las predicciones corresponden a:

$$b_x = (t_x) + c_x \quad (1)$$

$$b_y = (t_y) + c_y \quad (2)$$

$$b_w = p_w e^{tw} \quad (3)$$

$$b_h = Phe^{th} \quad (4)$$

RED DARKNET.- YOLOv3 sigue el principio de predicción de coordenadas en YOLOv2. Para predecir las categorías, se aplican etiquetas múltiples y clasificación

múltiple en lugar de la etiqueta única original y la clasificación múltiple [6,13,25]. La red YOLOv3-tiny básicamente puede satisfacer requisitos en tiempo real basados en recursos de hardware limitados [6,7,8]. YOLOv3-tiny crea una pirámide característica con una semántica fuerte a dos escalas mediante la adopción de capas de submuestreo y un enfoque de fusión. Como se muestra en la Figura 4, el tamaño de las dos escalas es 13x13 y 26x26, que se obtienen en la detección del objetivo de tamaño ordinario, respectivamente. Finalmente, dos escalas se fusionan al final.

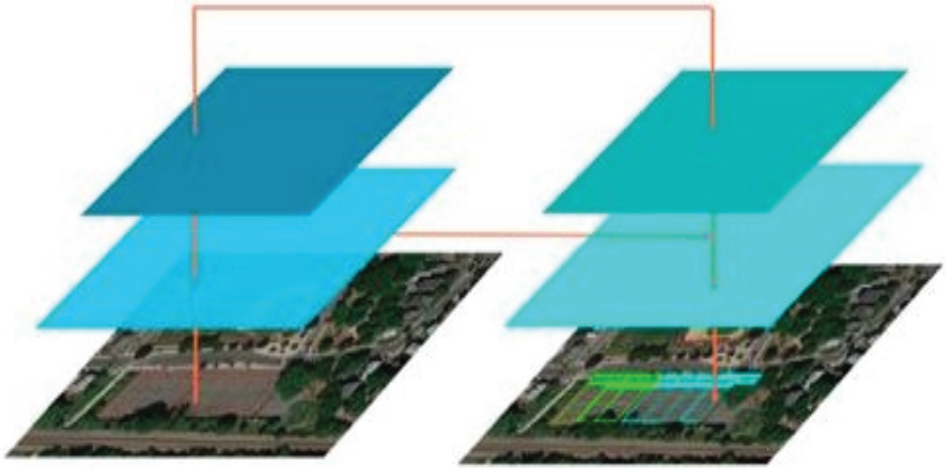

Figure 4

Multi-scale prediction in the YOLOv3-tiny network. Fuente: An improved incremental network for real-time object detection. Applied Sciences. 2019.

Se crea un entorno de desarrollo en Anaconda con las siguientes librerías, para el correcto funcionamiento de YOLOv3. OpenCV, NumPy, Sklean, Keras, Tensor flow, Cuda, Cudnn. En la figura 5 se muestra las pruebas iniciales de la correcta instalación de YOLOv3 en Windows 10 y Ubuntu 18.04.

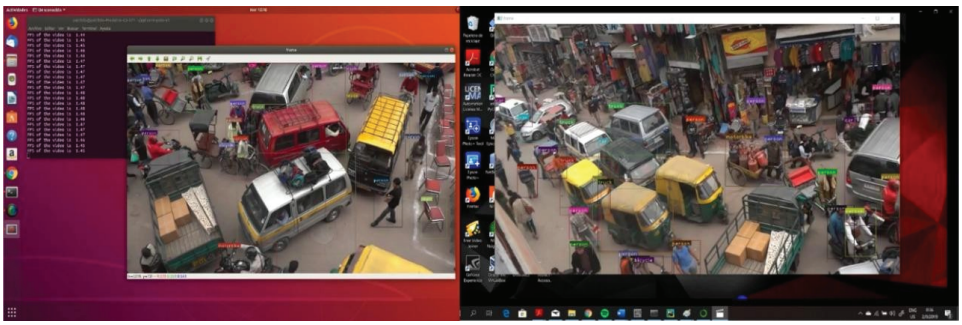

Figure 5

YOLOv3 Ubuntu 18.04 – Windows 10.

Para el seguimiento de objetos adaptaremos la librería Deep Sort a la versión de YOLOv3, el algoritmo de Deep Sort se lo muestra a continuación:

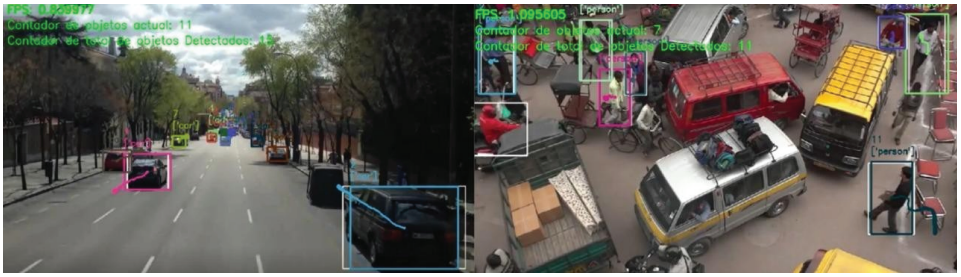

Figure 6

Detección de Autos y Peatones.

Matching Cascade

Input: Track indices T = {1, . . . N}, Detection indices D {1, . M}, Maximum age Amax l: Compute cost matrix C = [ct,J] Compute gate matrix B = [bt j] 3: Initialize set of matches M Ø 4: Initialize set of unmatched detections u D [an,j] K— min-cost-matching(C, 'Tn u) 8: M { (i, j) I bt,J > 0} 9: end for 10: return M, u.

3. Resultados y Discusión

Para la prueba se implementó un entorno en anaconda con todas las librerías necesarias para cada una de los sistemas operativos las pruebas de compilación se pueden observar en la figura 6 la detección de Autos y Peatones.

Como se muestra en la tabla 1 el rendimiento del algoritmo fue mejor en Ubuntu 18.04 LTS, con una media de 1,74 fps y en Windows se obtuvo un rendimiento de 1,03 fps.

Como se muestra en la tabla 1 el rendimiento del algoritmo fue mejor en Ubuntu 18.04 LTS, con una media de 1,74 fps y en Windows se obtuvo un rendimiento de 1,03 fps. Para la detección se implementó un algoritmo que es capaz de identificar al peatón y se calculó su distancia con respecto al vehículo, con esto se identifica si el peatón esta cruzando frente al vehículo, el funcionamiento del algoritmo se muestra en la figura 7.

Limitaciones: El algoritmo solo fue probado con videos de peatones y automóviles cargados directamente al input, este fue desplegado tanto en un entorno Windows y Linux, obteniendo mejores resultados en windows los resultados obtenidos los podemos ver en la tabla 1. La velocidad del algoritmo está ligada al hardware mismo que se describe en la sección III, con un mejor hardware se podría conseguir mejores resultados de FPS. No se ha realizado pruebas en un entorno real, por lo cual no se

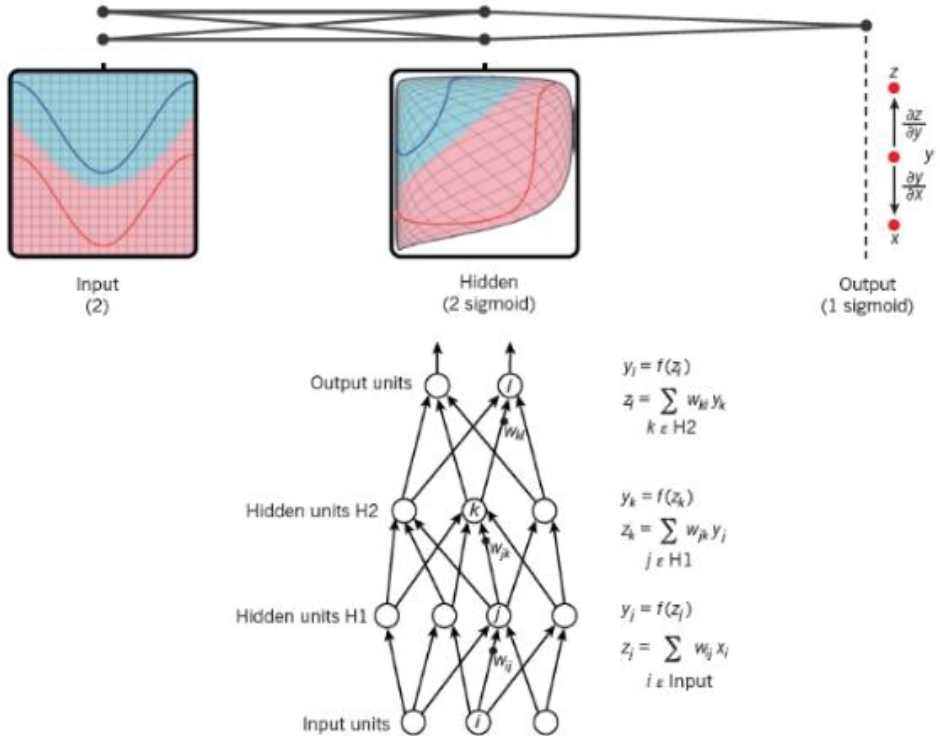

Figure 7

Detección de Autos y Peatones.

Table 1

FPS Promedio

FPS(MUESTRA)	WINDOWS	UBUNTU
1	1,06	1,77
2	1,09	1,88
3	0,97	1,67
4	0,94	1,69
5	1,03	1,71
6	1,09	1,75
7	1,06	1,45
8	1,03	1,85
9	0,98	1,76
10	1,02	1,83
PROMEDIO	1,03	1,74

puede concluir la eficiencia en tiempo real del algoritmo, esta se la prende de realizar en un próximo estudio con hardware especializado.

Figure 8

Detección de Cruze de Peatones.

3.1. Discusión

Aunque el modelo propuesto logró resultados experimentales satisfactorios, en la detección de peatones y vehículos. Las pruebas solo se las realizaron con videos de tránsito, para trabajos futuros se podría implementar este algoritmo a un sistema de cámaras de video vigilancia ya que este es un sistema real y se obtendría datos más reales de la efectividad del algoritmo.

4. Conclusiones

En la investigación, se implementó Deep Sort en la versión YOLOv3, para esto se utilizó los pesos pre-entrenados de YOLOv3, y se los convirtió a un modelo Keras. Se delimito la detección de YOLO a un solo objeto para nuestro caso experimental para peatones y vehículos. A continuación, se realizó un contador de objetos para identicar el número de objetos detectados, el modelo mostro una buena capacidad de detección tanto de vehículos como de peatones. Por lo tanto, la red propuesta es efectiva para la detección de peatones y vehículos.

El rendimiento de YOLOv3 fue mejor en Ubuntu 18.04 LTS, con una media de 1,74 fps y en Windows se obtuvo un rendimiento de 1,03 fps.

Agradecimientos

Agradecemos Universidad de Málaga por todos los conocimientos impartidos, a todas las personas que con su conocimiento hicieron posible este trabajo.

Conflicto de Intereses

No existe conflicto de intereses.

References

[1] Poppe R. A survey on vision-based human action recognition. Image and vision Computing. 2010 June; 28(6).

[2] Andres Felip JIM. Detección de flujo vehicular basado en visión artificial. Scientia et Technica. 2007; 3(35).

[3] Rusk N. Deep learning. Nature Methods. 2016; 13(1).

[4] Bewley A,GZ,OL,RF,&UB. Simple online and realtime tracking. In IEEE international conference on image processing (ICIP); 2016. p. 3464-3468.

[5] Human fall detection algorithm based on YOLOv3. In 2020 IEEE 5th International Conference on Image, Vision and Computing (ICIVC). In IEEE 5th International Conference on Image, Vision and Computing (ICIVC) ; 2020. p. 50-54.

[6] He W,HZ,WZ,LC,&GB(. TF-YOLO: An improved incremental network for real-time object detection. Applied Sciences. 2019; 9(16).

[7] Zhang F,LC,&YF. Vehicle detection in urban traffic surveillance images based on convolutional neural networks with feature concatenation. Sensors. 2019; 19(3).

[8] Wojke N,&BA. Deep cosine metric learning for person reidenti. In IEEE winter conference on applications of computer vision; 2018. p. 748-756.

[9] Gong J,ZJ,LF,&ZH. Vehicle detection in thermal images with an improved yolov3-tiny. In IEEE international conference on power, intelligent computing and systems (ICPICS); 2020. p. 253-256.

[10] Benjdira B,KT,KA,AA,&OK. Car detection using unmanned aerial vehicles: Comparison between faster r-cnn and yolov3. In 1st International Conference on Unmanned Vehicle Systems-Oman (UVS); 2019. p. 1-6.

[11] Huang YQ,ZJC,SSD,YCF,&LJ. Optimized YOLOv3 algorithm and its application in traffic flow detections. Applied Sciences. 2020; 10(9).

[12] Zhang H,QL,LJ,GY,ZY,ZJ,&XZ. Real-time detection method for small traffic signs based on Yolov3. IEEE Access. 2020; 8.

[13] Zhang FK,YF,&LC. Fast vehicle detection method based on improved YOLOv3. Computer Engineering and Applications. 2020; 55(2).

[14] Hassan, N. I., Tahir, N. M., Zaman, F. H. K., & Hashim, H. In 10th IEEE International Conference on Control System, Computing and Engineering (ICCSCE); 2020. p. 131-136.

[15] Ouyang L,&WH. Vehicle target detection in complex scenes based on YOLOv3 algorithm. In IOP Conference Series: Materials Science and Engineering; 2019. p. 052018.

[16] Wang X,WS,CJ,&WY. Data-driven based tiny-YOLOv3 method for front vehicle detection inducing SPP-net. IEEE Access. 2020; 8.

[17] Du L,CW,FS,KH,LC,&PZ. Real-time detection of vehicle and traffic light for intelligent and connected vehicles based on YOLOv3 network. In 5th International Conference on Transportation Information and Safety (ICTIS) ; 2019. p. 388-392.

[18] Zadobrischi E,&NM. Pedestrian detection based on TensorFlow YOLOv3 embedded in a portable system adaptable to vehicles. In International Conference on Development and Application Systems (DAS); 2020. p. 21-26.

[19] Zhao S,&YF. Vehicle detection based on improved yolov3 algorithm. In International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS) ; 2020. p. 76-79.

[20] Zhou L,LJ,&CL. Vehicle detection based on remote sensing image of Yolov3. In IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC); 2020. p. 468-472.

[21] Hassan NI,TNM,ZFHK,&HH. People detection system using YOLOv3 algorithm. In IEEE International Conference on Control System, Computing and Engineering (ICCSCE); 2020. p. 131-136.

[22] Zhao H,ZY,ZL,PY,HX,PH,&C. Mixed YOLOv3-LITE: a lightweight real-time object detection method. Sensors. 2020; 20(7).

[23] Pérez RM,AJS,&PAM. Introducción al Aprendizaje Automático con YOLO. 2019; 3(6).

[24] Wojke N,BA,&PD. Simple online and realtime tracking with a deep association metric. In IEEE International Conference on Image Processing (ICIP); 2017. p. 3645-3649.

[25] Zhang X,&ZX. An efficient and scene-adaptive algorithm for vehicle detection in aerial images using an improved YOLOv3 framework. ISPRS International Journal of Geo-Information. 2019; 8(11).